WHAT IS CLAIMED IS:

1. A decoding method comprising:

(a) extracting a preselected number of bits from a bitstream to be decoded;

(b) accessing a table entry of a table using an index generated in response to the extracted bits ; and

(c) if a first value in the table entry indicates a second value in the table entry comprises a decoded result, retrieving said decoded result.

2. The method of claim 1 further comprising:

(d) if said first value in the table entry indicates a second value in the table entry does not comprise a decoded result, accessing a next table entry in said table in response to said second value.

3. The method of claim 1 wherein the index comprises a sum of the value of the extracted bits and a base address of said table.

4. The method of claim 2 wherein if said first value indicates said second value does not comprise a decoded result, said second value comprises a first portion and a second portion, the first portion of said second value comprising an offset, and said second portion comprising an additional number of bits to extract from said bitstream.

5. The method of claim 4 further comprising:

(e) offsetting a current index by the first portion of the second value;

(f) extracting a number of bits from said bitstream equal to the second portion of the second value; and

(g) offsetting the current index by a value of the extracted bits equal to the second portion of the second value to generate a next pointer into the table.

6. The method of claim 5 further comprising:

(h) accessing the table using the pointer generated in step (g);

(i) if a first value in the entry accessed in step (h) indicates a second value in the entry comprises a decoded result, retrieving said decoded result; and

(j) if said first value in the entry accessed in step (h) indicates a second value in the entry does not comprise a decoded result, accessing a next entry in said table in response to said second value.

7. The method of claim 6 further comprising:

(k) repeating steps (e)-(j) until all bits in said bitstream are decoded.

8. A signal processing system comprising:

an input port configured to receive an encoded digital signal; and

a processor including a decoder,

memory, including a table having a plurality of entries, each entry having a first portion and a second portion, wherein the decoder is coupled to the table, and wherein said decoder comprises:

logic for extracting a preselected number of bits in a bitstream of said encoded digital signal.

logic for addressing an entry in said table in response to the preselected number of bits; and

logic for retrieving a decoded value if a value the first portion of an addressed entry indicates a value in the second portion of the addressed entry comprises a decoded result.

9. The system of claim 8 wherein said logic for addressing an entry in said table further comprises logic for addressing a next entry in said table if the first value in the entry indicates the second value does not comprise a decoded result.

10. The system of claim 9 wherein if said first value indicates said second value does not comprise a decoded result, said second value comprises a first portion and a second portion, the first portion of said second value comprising an offset value, and said second portion comprising an additional number of bits to extract from said bitstream.

11. The system of claim 10 wherein said decoder further comprises:

logic for offsetting a current index by the first portion of the second value; and logic for extracting a next number of bits from said bitstream equal to the value in the second portion of the second value.

12. The system of claim 11 further comprising logic generating a next pointer into the table, the logic including logic for further offsetting the current index by a value of the next number of bits extracted from said bitstream .

13. The system of claim 12 wherein the decoder further comprises:

logic for accessing the table using said next pointer; and

logic for repeating the operations of:

retrieving a decoded value if a value of the first portion of an addressed entry indicates a value in the second portion of the addressed entry comprises a decoded result; and

addressing a next entry in said table if the first value in the entry indicates the second value does not comprise a decoded result.

14. A method for generating a decode table comprising:

generating a first table having a plurality of entries, each entry including a plurality of fields, a first field comprising a value selectively representing a code word of a plurality of code words in a codetable or a portion of a code word of said plurality of code words, a second field comprising a length of a code word, and a third field comprising a difference of a length of a codeword and a value for selectively shifting a code word, wherein a shifted codeword is matched against a value in the first field of one or more entries in the first table; and

generating a second table using values in the entries of the first table, the second table operable for decoding a bitstream encoded using said codetable, the step of generating the second table comprising:

if a value in the first field in an entry of the first table corresponds to a code word and a value in the second field of the entry corresponds to the length of the code word in the first field in the entry of the first table, setting a result tag in a first portion of an entry of the second table to a first value, and setting a decode result corresponding to the code word in a second portion of the entry of the second table.

15. The method of claim 14 wherein the first value of the result tag is operable for determining that the value in the second field represents a decode result.

16. The method of claim 15 further comprising:

if the value in a first field in an entry of the first table does not correspond to a code word or the value in the second field of the entry does not correspond to a length of a code word in a first field in the entry of the first table, and if the value in the second field of the entry in the first table equals the value in the third field of a next entry of the first table and the value in the first field of the entry in the first table equals the difference of the value in the second and third fields of the next entry in the first table right-shifted by the value in the first field of the next entry of the first table:

(a) setting the result tag in a first field in an entry of the second table to a second result value;

(b) setting a difference of an index of the next entry of the first table and an index of the entry of the second table in a third portion of the entry of the second table; and

(c) setting the difference of the value in the second and third fields of the next entry in the first table in a fourth portion of the entry of the second table.

17. The method of claim 16 wherein the second value of the result tag is operable for determining that the value in the third portion represents an index offset and the value in the fourth portion represents a number of bits to extract from the bitstream.

18. The method of claim 17 wherein the third and fourth portion of the entry of the second table comprise subfields of a second field of the second table.